

REMARKS

Claims 41-80 stand rejected. Claims 42, 45-47, 49, 51-53, 55, 57, 59-61, 63, 65-67, 69, 72, 73 and 75-80 have been amended. Claims 41, 43, 44, 56, 58, 68, 70 and 71 have been canceled. New claims 81-86 have been added. The Applicants respectfully request reconsideration in view of the foregoing amendments. No new matter has been added.

Claim Objections

Claim 68 was objected to because it was improperly labeled “previously presented” when the claim was amended. Claim 68 has been canceled. The objection is now moot.

Claim Rejections – 35 U.S.C. § 112

Claims 41-80 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite. Specifically, with respect to claims 41-55 and 68-80, the Office Action states that the term “performing the activity” is unclear as to which activity is referred. Similarly, with respect to claims 56-67, the Office Action states that the term “the activity” is also not clear as to which activity is referred. Claims 41, 56 and 68 have been canceled. This rejection is now moot.

Claim Rejections – 35 U.S.C. § 103

Claims 41-80 were rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,308,163 to Du et al. (hereinafter “Du '163”) in view of U.S. Patent 6,041,306 to Du et al. (hereinafter “Du '306”) and in further view of U.S. Patent 5,522,070 to Sumimoto (hereinafter “Sumimoto”). Independent claims 41, 56 and 68 have been canceled and replaced with new claims 81, 83 and 85 respectively.

With respect to claim 81, an automated method is recited for processing a workflow among a plurality of activity servers. The method comprises, at each activity server, the steps of: (i) obtaining workflow transition information from a common database, the workflow transition information defining a sequence of a plurality of activities for a workflow; (ii) retrieving a workflow packet from a workflow queue maintained in the common database, the workflow packet corresponding to an activity defined in the sequence of activities for the workflow; (iii) performing the activity corresponding to the retrieved workflow packet; (iv) determining a next activity in the sequence of activities for the workflow from the workflow transition information; and (iv) wherein the next activity is capable of being performed by the activity server, performing the next activity without requesting a next workflow packet from the workflow queue. Similar features are recited in new claims 83 and 85, respectively. Support for new claim 81, 83 and 85 can be found in FIGS. 3, 4 and the subject specification as originally filed on page 10, line 22 to page 11, line 13, page 13, line 11 to page 15, line 10, page 16, line 4 to page 19, line 14.

Accordingly, the invention as recited in claims 81, 83 and 85 enables serial processing of a workflow packet by a plurality of activities within a single transaction upon a single activity server. See subject specification as originally filed on page 18, lines 21-23. For example, in one embodiment, an activity server retrieves from the workflow queue a single workflow packet corresponding to one activity, performs that activity and continues to perform additional consecutive activities as defined in the workflow transition information until the server determines that it cannot perform the next activity in the workflow. If the activity server is incapable of performing the next activity in the workflow, the activity server forms the next workflow packet corresponding to the next activity and forwards the next workflow packet to the

workflow queue for retrieval by another activity server capable of performing the next activity. See subject specification as originally filed on page 16, line 4 to page 19, line 14. In another embodiment, the activity server retrieves a plurality of workflow packets corresponding to a sequence of activities that the activity server is capable of performing and performs each activity until the server determines that it cannot perform the next activity in the workflow. See subject specification as originally filed on page 10, line 22 to page, 11, line 13 and page 13, line 11 to page 15, line 10.

Neither Du '163, Du '306, nor Sumimoto, singly or in combination, teach or suggest serial processing of a workflow packet by a plurality of activities within a single transaction upon a single activity server as now recited in claims 81, 83 and 85, respectively. Specifically, none of the cited art of record teaches or suggests having each of a plurality of activity servers, which together process a workflow, being operable (i) retrieve a workflow packet from a workflow queue maintained in the common database, (ii) determine a next activity in the sequence of activities for the workflow from the workflow transition information; and (iii) wherein the next activity is capable of being performed by the activity server, perform the next activity without requesting a next workflow packet from the workflow queue.

Moreover, none of the cited references teach or suggest that where the next activity is incapable of being performed by the activity server, the activity server forms another workflow packet corresponding to the next activity and forwards the next workflow packet to the workflow queue for retrieval by another activity server capable of performing the next activity, as recited in new claims 82, 84 and 86. Support for these new claims can be found at least in the subject specification as originally filed on page 4, lines 10-12.

In contrast, Du '163 discusses a multi-level resource manager hierarchy that represents an enterprise wide view of resource capabilities. Automated resource managers at each level communicate with each other through a series of messages to delegate, refer and request resources for use in a workflow processing. (See Du '163: Abstract)

Du '306 merely discusses a workflow process software engine that executes a workflow process defined as a directed graph including an interconnected set of so-called "workflow nodes" and "rule nodes." Each workflow node corresponds to a process activity, and each rule node represents a decision point in the process flow. For each workflow node, the workflow engine executes the associated process activity causing the process activity (task) to be mapped to an appropriate resource. (See Du '306: col. 7, line 36 to col. 9, line 4; col. 11, line 26 to col. 12, line 67)

In Sumimoto, a client scheduler merely decides how to distribute processes among multiple computers in a network. (See Abstract).

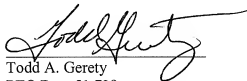
For at least these reasons, claims 81-86 are patentable, as they are neither anticipated by or obvious in view of the cited art of record.

Furthermore, by virtue of at least their dependency upon claims 81, 83 and 85, respectively and the additional features recited therein, claim 42, 45-55, 57, 59-67, 69, 72-80, 82, 84 and 86 are also patentable.

CONCLUSION

In view of the above amendments and remarks, it is believed that claims 42, 45-55, 57, 59-67, 69, 72-86 are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned.

Respectfully submitted,



Todd A. Gerety
PTO Reg. 51,729
Attorney for the Applicants
Proskauer Rose LLP
One International Place
Boston, MA 02110

Date: January 24, 2008

Tel. (617) 526-9655
Fax (617) 526-9899